Average-case analysis of perfect sorting by reversals

Mathilde Bouvel, Cédric Chauve, Marni Mishna, Dominique Rossin

Average-case analysis of perfect sorting by reversals

Mathilde Bouvel[*], Cedric Chauve[†], Marni Mishna[†], Dominique Rossin[‡]

Abstract

Perfect sorting by reversals, a problem originating in computational genomics, is the process of sorting a signed permutation to either the identity or to the reversed identity permutation, by a sequence of reversals that do not break any common interval. Bérard *et al.* (2007) make use of strong interval trees to describe an algorithm for sorting signed permutations by reversals. Combinatorial properties of this family of trees are essential to the algorithm analysis. Here, we use the expected value of certain tree parameters to prove that the average run-time of the algorithm is at worst, polynomial, and additionally, for sufficiently long permutations, the sorting algorithm runs in polynomial time with probability one. Furthermore, our analysis of the subclass of commuting scenarios yields precise results on the average length of a reversal, and the average number of reversals.
A preliminary version of this work appeared in the proceedings of Combinatorial Pattern Matching (CPM) 2009, Lectures Notes in Computer Science, vol. 5577, pp. 314–325, Springer.

1 Introduction

There are many examples where the average case complexity of a sorting algorithm is neatly computed with a generating function computation on a related family of trees. Most of the heavy lifting is done by complex analysis. We give a new example here: we perform an average case analysis of a sorting algorithm from computational genomics by generating function analysis of a family of trees.

Motivation: a computational genomics problem. With the availability of a growing number of sequenced and assembled genomes, the comparison of whole genomes in terms of large-scale evolutionary events called *genome rearrangements* is a fundamental task in computational genomics. Computing a genomic distance and/or a parsimonious evolutionary scenario between a pair of genomes is one of the basics problems in this field, with applications such as reconstructing phylogenies [25] or unraveling evolutionary properties of groups of genomes [24, 26]. This general problem was formally introduced as an algorithmic problem by Sankoff in [27]. Since then, these questions have been extensively investigated, for different models of genomes and genome rearrangements, leading to a rich corpus of combinatorial and algorithmic results; we refer the reader to the recent book by Fertin *et al.* on this topic [15].

Signed permutations, reversals and scenarios. In this work, we study the problem of computing parsimonious perfect reversal scenarios between unichromosomal genomes. Unichromosomal genomes can be modeled by signed permutations: each element of a permutation corresponds to a genomic marker (a gene for example but not exclusively), defined as a segment of the double-stranded DNA molecule forming a chromosome segment, with its sign indicating which strand of the chromosome carries the marker.

A reversal is an evolutionary event that reverses a chromosomal segment. It can be modeled as a discrete operator acting on a signed permutation, reversing the order and sign of an interval of the permutation. A sequence of reversals that transforms one signed permutation into another one is viewed as a possible

[*]CNRS, Université de Bordeaux I, LaBRI, Bordeaux, France
[†]Department of Mathematics, Simon Fraser University, Burnaby (BC), Canada, V5A 1S6
[‡]CNRS, École Polytechnique, LIX, Palaiseau, France

evolutionary scenario from a genome to another one. Such a scenario is said to be *parsimonious* if no other scenario exists that requires less reversals.

Notice that up to relabeling, we can always assume that one of the two permutations is the identity. Without loss of generality, we assume that the permutation we want to obtain at the end of a scenario is the identity, hence the connection with the sorting problem. Sankoff initiated in [27] the algorithmic study of parsimonious reversal scenarios. Since then this problem has been considered by many authors, and efficient algorithms exist to compute a parsimonious scenario [19, 8, 30].

Common intervals and perfect scenarios. However, there can be many scenarios that satisfy this parsimony constraint. In fact, on real data sets, there can be an exponential number of parsimonious reversal scenarios (see [10] for example). This illustrates the need to refine the criteria which defines a good evolutionary scenario, and to go beyond the simple parsimony criterion. This need motivates the introduction of the *perfect scenarios*.

Perfect scenarios aim at avoiding convergent evolution. That is, if groups of genes or other genomic markers are co-localized in genomes of two species, a preferred scenario would preserve this quality back to the ancestral genome; the group of genes should remain together in every step of the evolution. In the combinatorial model based on signed permutations, this appears as a *common interval*: a collection of sequential numbers that forms an interval both in the identity permutation and (in absolute values) in the signed permutation to be sorted. A sorting scenario for a signed permutation σ is said to *break* a common interval I of σ if it contains a reversal such that the elements of I do not form an interval in the permutation obtained after the reversal is performed. A scenario that does not break any common interval is said to be *perfect*, and may very well be longer than the shortest, purely parsimonious, scenario. However it is considered to have stronger properties as a hypothetical evolutionary scenario. The algorithmic problem is thus stated:

> Given a signed permutation, compute a sequence of reversals that sorts it towards the identity or reversed identity, does not break any of its common intervals, and is shortest among all such scenarios.

Notice that the permutation obtained at the end of the scenario can be the identity, or the reversed identity, which represents the same genome but viewed from the other end.

Computing perfect scenarios: existing results. The refined problem that asks to preserve only a predefined subset of the existing common intervals is NP-complete [16]. Even in the general problem, which considers all common intervals, no algorithms with polynomial worst-case time complexity are known. However, some fixed parameter tractable (FPT) algorithms have been described [4, 6].

There also exists some classes of signed permutations that define tractable instances [3, 4, 13]. Among such tractable classes of signed permutations, *commuting permutations* is the sub-class of signed permutations that can be sorted by a *commuting scenario*, *i.e.* by a perfect scenario with the striking trait that the property of being a perfect scenario is preserved even when the sequence of reversals is reordered in every possible way. Surprisingly, examples of commuting scenarios arise in the study of mammalian genome evolution [3].

A link with trees and its applications: new results. The central combinatorial object in the theory of perfect sorting by reversals is the "strong interval tree" which tracks all common intervals of a (signed) permutation. It serves as a guide for the computation of perfect scenarios and the parameters introduced in the FPT algorithms described in [4, 6] read naturally in terms of this tree. This link opens the way to a refined analysis of some of the existing algorithms for perfect sorting by reversals, which is the purpose of our work.

The two key new results in Section 3 are Theorem 10, which states that for large enough n, with probability 1, computing a perfect scenario for signed permutations can be done in time polynomial in n and Theorem 15, which states that computing a perfect scenario can be done in polynomial time on average. Section 4 offers two new results on the average shape of a commuting scenario: we show that in parsimonious perfect scenarios for commuting permutations of size n, the average number of reversals is asymptotically $1.2n$, and the average length of a reversal is asymptotically $1.05\sqrt{n}$.

We conclude by discussing the relevance of these results, both from theoretical and applied point of views, and outlining future research.

2 Preliminaries

We first summarize the combinatorial and algorithmic frameworks for perfect sorting by reversals. For a more detailed treatment, in particular for properties of the strong interval tree, we refer the reader to [4].

Permutations, reversals, common intervals and perfect scenarios. A *signed permutation* of size n is a permutation of the set of integers $\{1, 2, \ldots, n\}$ in which each element additionally has a sign, either positive or negative. For clarity, negative integers are represented by placing a bar over them and positive signs are omitted. We write our permutations in one line notation. For example, $\sigma = [1\ \overline{3}\ \overline{2}\ 5\ 4\ 6]$ is a signed permutation of size 6. We denote by Id_n (resp. $\overline{Id_n}$, Id_n^m) the identity (resp. reversed identity, mirrored identity) permutation, $[1\ 2 \ldots n]$ (resp. $[\overline{n}\ \ldots\ \overline{2}\ \overline{1}]$, $[n\ \ldots\ 2\ 1]$). When the number n of elements is clear from the context, we will simply write Id, $\overline{Id}$, or Id^m.

An *interval* I of a signed permutation σ of size n is a segment of adjacent elements of σ. The *content* of I is the subset of $\{1, \ldots, n\}$ defined by the absolute values of the elements of I. Given σ, an interval is defined by its content and from now, when the context is unambiguous, we identify an interval with its content.

The *reversal* of an interval of a signed permutation reverses the order of the elements of the interval, while changing their signs. The length of a reversal is the number of elements in the interval that is reversed. If σ is a permutation, we denote by $\overline{\sigma}$ the permutation obtained by reversing the complete permutation σ. A *scenario* for σ is a sequence of reversals that transforms σ into Id_n or $\overline{Id_n}$. The *length* of such a scenario is the number of reversals it contains. A scenario of minimal length is a *parsimonious scenario*.

Example 1. Let $\sigma = [1\ \overline{4}\ \overline{5}\ 2\ \overline{3}\ 6]$ be a signed permutation of size 6, then $\overline{\sigma} = [\overline{6}\ 3\ \overline{2}\ 5\ 4\ \overline{1}]$. Reversing, in σ, the interval $[\overline{5}\ 2\ \overline{3}]$, or equivalently the set $\{2, 3, 5\}$, yields the signed permutation $[1\ \overline{4}\ 3\ \overline{2}\ 5\ 6]$. Reversing successively $\{2, 3, 4\}$ and $\{3\}$ completes this first reversal to form a parsimonious scenario of length 3.

A *common interval* of a permutation σ of size n is a subset of $\{1, 2, \ldots, n\}$ that is an interval in both σ and the identity permutation Id_n. The singletons and the set $\{1, 2, \ldots, n\}$ are always common intervals called *trivial common intervals*.

Example 2. The common intervals of $\sigma = [1\ \overline{3}\ \overline{2}\ 5\ 4\ 6]$ are $\{2, 3\}$, $\{1, 2, 3\}$, $\{4, 5\}$, $\{4, 5, 6\}$, $\{2, 3, 4, 5\}$, $\{2, 3, 4, 5, 6\}$, $\{1, 2, 3, 4, 5\}$, $\{1, 2, 3, 4, 5, 6\}$, and the singletons $\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}$.

Two distinct sets (intervals here) I and J *commute* if their contents trivially intersect, that is either $I \subset J$, or $J \subset I$, or $I \cap J = \emptyset$. If intervals I and J do not commute, they *overlap*. A scenario S for σ is a *perfect scenario* if no reversal of S breaks any common interval of σ, or equivalently [4] if every reversal of S commutes with every common interval of σ. It is easy to see that there always exists a perfect scenario for a given signed permutation. A perfect scenario of minimal length, among all perfect scenarios, is a *parsimonious perfect scenario*.

A permutation σ is said to be *commuting* if there exists a scenario for σ such that for every pair of reversals the corresponding intervals commute. Such a scenario is called a *commuting scenario* and is obviously perfect. It was shown in [4] that, if a signed permutation can be sorted by a commuting scenario, then any other perfect scenario for this signed permutation has the same set of reversals, and conversely every reordering of the reversals also gives a perfect scenario. This implies that a commuting scenario is also a parsimonious perfect scenario.

Example 3. Let $\sigma = [1\ \overline{3}\ \overline{2}\ 5\ 4\ 6]$ be a signed permutation of size 6. The scenario $\{2, 3\}$, $\{4, 5\}$, $\{4\}$, $\{5\}$ is a commuting scenario, and σ is a commuting permutation.

Remark 4. Commuting permutations have been investigated, in connection with permutation patterns, under the name of *separable* permutations [21].

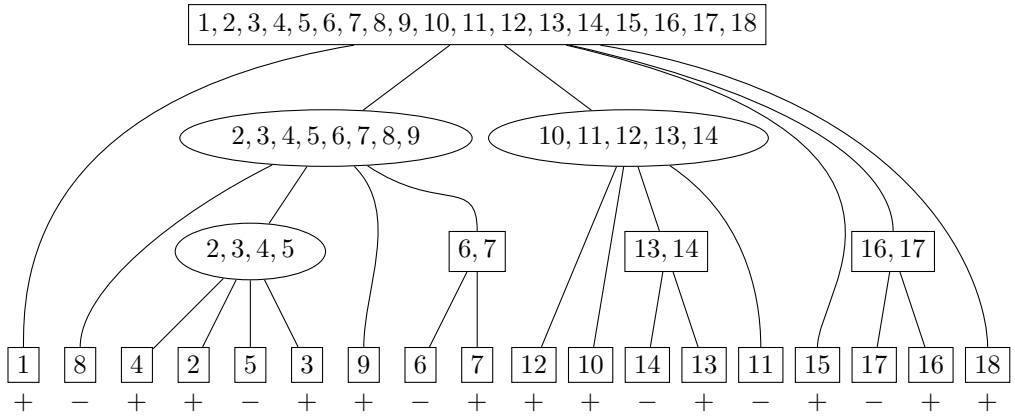

Figure 1: *The strong interval tree* $\mathbf{T}([1\ \overline{8}\ 4\ 2\ \overline{5}\ 3\ 9\ \overline{6}\ 7\ 12\ 10\ \overline{14}\ 13\ \overline{11}\ 15\ \overline{17}\ 16\ 18])$. *Vertices are labeled by the strong intervals. There are three non-trivial linear vertices (rectangular) and three prime vertices (round). The root and the vertex* $\{6, 7\}$ *are increasing linear vertices, while the linear vertices* $\{16, 17\}$ *and* $\{13, 14\}$ *are decreasing.*

The strong interval tree. First, we remark that the following definitions are valid for both signed and unsigned permutations. A common interval I of a permutation σ is a *strong interval* of σ if it commutes with every other common interval of σ. The inclusion order on the set of strong intervals of a permutation of size n defines an n-leaf tree, denoted by $\mathbf{T}(\sigma)$, whose leaves are the singletons and whose root is the interval containing all elements of the permutation. We require that the elements of $\{1, 2, \ldots, n\}$ appear on the leaves of $\mathbf{T}(\sigma)$ from left to right in the same order they do in σ. This implies that the children of every internal vertex of $\mathbf{T}(\sigma)$ are totally ordered, or in other words that $\mathbf{T}(\sigma)$ is a plane tree *i.e.* a tree embedded in the plane. We identify a vertex of $\mathbf{T}(\sigma)$ with the strong interval it represents. If σ is a signed permutation, the sign of every element of σ is given to the corresponding leaves in $\mathbf{T}(\sigma)$. Figure 1 shows an example of a strong interval tree.

Let I be a strong interval of σ that is not a singleton and let $\mathcal{I} = (I_1, \ldots, I_k)$ the unique partition of the elements of I into maximal strong intervals, from left to right. The *quotient permutation* of I, denoted σ_I, is the permutation of size k defined as follows: $\sigma_I(i)$ is smaller than $\sigma_I(j)$ in σ_I if and only if any element of the content of I_i is smaller than any element of the content of I_j. A fundamental property of the strong interval tree is that the quotient permutation σ_I of an internal vertex I having k children ($k \geq 2$) in a strong interval tree can only be either Id_k, Id_k^m, or a permutation of size k, with $k \geq 4$, whose only common intervals are the $k + 1$ trivial common intervals. Such a permutation with no non-trivial common interval is called a *simple permutation*. The shortest simple permutations are of size 4 and are [3 1 4 2] and [2 4 1 3]. We describe simple permutations in more detail in Section 3.1.

For an internal vertex I, if $\sigma_I = Id_k$ (resp. Id_k^m, is a simple permutation), then I is said to be an *increasing linear* vertex (resp. *decreasing linear* vertex, *prime* vertex). Another crucial property of a strong interval tree is that no two increasing (resp. decreasing) linear vertices can be adjacent: if a linear vertex is the child of another linear vertex, then one of them is increasing and the other one is decreasing.

The strong interval tree is also known as the *substitution decomposition tree* [1], and is very similar to *PQ-trees* [9], a data structure used to represent the common intervals of two or more permutations [20, 7, 11]. More precisely, the strong interval tree defines a PQ-tree if linear (resp. prime) vertices are called Q-vertices (resp. P-vertices). This PQ-tree can be computed in linear time [7]. To obtain the strong interval tree, the quotient permutation of each vertex needs then to be computed. The algorithm of [7] can be adapted to compute them, still in linear time. Indeed, given the tree, the quotient permutations can be computed as follows: consider the elements on the leaves, from 1 to n, and propagate these elements along the edges of the tree towards the root, until a previously used edge is encountered. The relative ordering of the elements at every internal vertex of the tree gives the quotient permutation, and their computation is obtained in $\mathcal{O}(n)$ time.

The strong interval tree as a guide for perfect sorting by reversals. The algorithm in [4] computing a parsimonious perfect scenario for a given signed permutation is the central object of study here, and is henceforth labeled Algorithm BBCP07.

To compute a parsimonious perfect scenario for a signed permutation σ, Algorithm BBCP07 heavily relies on the strong interval tree $\mathbf{T}(\sigma)$ of σ. It starts with computing this tree, and then assign signs to internal vertices according to the following rules: an increasing (resp. decreasing) linear vertex is signed $+$ (resp. $-$) and a prime vertex having a linear parent inherits its sign from its parent. Some prime vertices may remain unsigned at this step, and the algorithm will explore all the possible assignments of signs to these prime nodes. If p denote the number of prime vertices in $\mathbf{T}(\sigma)$, there may be up to 2^p possible assignments. The key ingredient of the algorithm is that any reversal in a perfect scenario is either a strong interval (hence a vertex of $\mathbf{T}(\sigma)$) or the union of consecutive children of a prime vertex of $\mathbf{T}(\sigma)$ [4, Proposition 2]. Hence a scenario can be computed by looking successively at each vertex of the strong interval tree, sorting a signed permutation, defined from its quotient permutation and the signs of its children, towards either the identity (if the vertex has sign $+$) or the reversed identity (if the vertex has sign $-$). More precisely, for each assignment of signs, a scenario is computed as follows:

- Transform the quotient permutation of each vertex into a signed permutation by lifting the sign of each child onto the corresponding element in the quotient permutation.
- For each prime node signed $+$ (resp. $-$) whose signed quotient permutation is τ (a signed permutation of size k), compute a parsimonious scenario from τ to Id_k (resp. to $\overline{Id_k}$). This is achieved using a polynomial-time algorithm solving the general sorting by reversal problem (without the 'perfectness' condition). The most efficient algorithm so far is the one of [30], that runs in $\mathcal{O}(k\sqrt{k \log k})$ time.
- In addition to the reversals obtained at the previous step, perform a reversal for every interval of σ that correspond to a vertex (internal vertex or leaf) in $\mathbf{T}(\sigma)$ whose parent is linear and whose sign is different from the sign of its parent.

The scenarios thus obtained are all perfect scenarios, and among them, those of minimal length are parsimonious perfect scenarios. For the correctness and complexity analysis of Algorithm BBCP07, we refer to [4].

Example 5. *On the example of Figure 1, the root of $\mathbf{T}(\sigma)$, its two prime children and vertex $\{6,7\}$ are signed $+$, whereas vertices $\{13,14\}$ and $\{16,17\}$ are signed $-$. For vertex $\{2,3,4,5\}$, the two possible sign assignments have to be tested. Choosing sign $+$ (resp. $-$) produces a scenario with 15 (resp. 14) reversals, among which 4 correct a sign mismatch between a vertex and its linear parent (for vertices $\{6\},\{13\},\{16\}$ and $\{16,17\}$) and the remaining 11 (resp. 10) arise from reversals in prime nodes. More precisely, sorting the right-most prime child of the root requires 3 reversals (through the optimal scenario $[\mathbf{3\,1\,\overline{4}\,\overline{2}}] \to [\mathbf{4\,\overline{1}\,\overline{3}\,\overline{2}}] \to [1\,\overline{\mathbf{4}}\,\overline{\mathbf{3}}\,\overline{\mathbf{2}}] \to [1\,2\,3\,4]$); when sign $+$ is chosen, the left-most prime child of the root is sorted in 4 reversals ($[\overline{\mathbf{3}}\,\mathbf{1}\,4\,2] \to [\overline{\mathbf{1}}\,\mathbf{3}\,\mathbf{4}\,2] \to [\overline{\mathbf{4}}\,\overline{\mathbf{3}}\,\mathbf{1}\,\mathbf{2}] \to [\overline{\mathbf{2}}\,\overline{\mathbf{1}}\,3\,4] \to [1\,2\,3\,4]$) and its prime child in 4 reversals ($[3\,1\,\overline{\mathbf{4}}\,\mathbf{2}] \to [\mathbf{3}\,\mathbf{1}\,\overline{2}\,4] \to [\overline{1}\,\overline{\mathbf{3}}\,\overline{\mathbf{2}}\,4] \to [\overline{\mathbf{1}}\,2\,3\,4] \to [1\,2\,3\,4]$); and when sign $-$ is chosen, the left-most prime child of the root is sorted in 3 reversals ($[\overline{3}\,\overline{\mathbf{1}}\,\mathbf{4}\,2] \to [\overline{3}\,\overline{\mathbf{4}}\,\mathbf{1}\,\mathbf{2}] \to [\overline{\mathbf{3}}\,\overline{\mathbf{2}}\,\overline{\mathbf{1}}\,4] \to [1\,2\,3\,4]$) and its prime child in 4 reversals ($[3\,1\,\overline{\mathbf{4}}\,2] \to [\mathbf{3}\,\mathbf{4}\,\overline{1}\,2] \to [\overline{4}\,\overline{3}\,\overline{\mathbf{1}}\,\mathbf{2}] \to [\overline{4}\,\overline{3}\,\overline{2}\,\mathbf{1}] \to [\overline{4}\,\overline{3}\,\overline{2}\,\overline{1}]$). Therefore, for the signed permutation σ of Figure 1, the length of a parsimonious perfect scenario is 14.*

The following proposition is a summary of some of the key results of [4] on Algorithm BBCP07, that will play a central role in our work.

Proposition 6 (Bérard *et al.* [4])**.** *Let σ be a signed permutation of size n. Let $\mathbf{T}(\sigma)$ be its strong interval tree, and denote by p its number of prime nodes. Then the followings are true:*

1. *Algorithm BBCP07 compute a parsimonious perfect scenario for σ in worst-case time $\mathcal{O}(2^p\, n\sqrt{n \log n})$;*
2. *σ is a commuting permutation if and only if $p = 0$;*
3. *if σ is a commuting permutation, then a sorting scenario for σ is perfect if and only if it consists of one reversal for every interval corresponding to a vertex of $\mathbf{T}(\sigma)$ that has a sign different from its parent.*

Hence it appears that prime vertices of the strong interval tree are fundamental in the exponential worst-case behavior of Algorithm BBCP07, and more generally in the hardness of the problem of perfect sorting by reversals. Indeed, an interpretation of the hardness result given in [16] in terms of strong interval tree is that perfect sorting by reversals is NP-complete for signed permutations whose strong interval tree contains only prime nodes.

3 On the number of prime vertices

As we shall soon see, the average-time complexity of Algorithm BBCP07 can also be bounded with the aid of strong interval trees. We use enumerative results on simple permutations to determine the "average shape" of a tree with n leaves. This average shape is extremely simple and has a single prime node. From this we can easily bound the average-time complexity.

3.1 Combinatorial preliminaries: strong interval trees and simple permutations

The following formal description of the underlying structure of the strong interval trees is useful for our enumerative analysis.

Definition 7. Let $\mathcal{T}_n$ be the family of plane trees satisfying the following properties:

P1. each tree has n leaves (n is the *size* of the trees of $\mathcal{T}_n$);

P2. each leaf is labeled by $+$ or $-$;

P3. the children of each internal vertex are totally ordered;

P4. each internal vertex has at least two children;

P5. if an internal vertex has k children, it is labeled either by Id_k, or Id_k^m, or a simple permutation of size k if $k \geq 4$;

P6. no edge is incident to two vertices labeled by Id or two vertices labeled by Id^m.

We previously noted that each permutation corresponds to a strong interval tree. We prove next that this correspondence is bijective.

Theorem 8. *There is a bijection between the set of signed permutations of size n and $\mathcal{T}_n$.*

Proof. First, it is immediate to see that a unique tree of $\mathcal{T}_n$ can be obtained from a signed permutation σ of size n. Indeed, it is enough to modify its strong interval tree $\mathbf{T}(\sigma)$ by labeling each leaf representing an element of σ by its sign, and each internal vertex corresponding to a strong interval I by the quotient permutation σ_I.

To get a signed permutation σ_T from a tree T of $\mathcal{T}_n$, we assign signed integers to its leaves and σ_T will be obtained by reading the leaves from left to right. The absolute values of the integers labeling the leaves are obtained by a top-down approach. We first assign the set of integers $I = \{1, \dots n\}$ to the root, together with a variable m set to 1 indicating the minimal value of I. We propagate this assignment from the root to the leaves as follows. Consider a node labeled by a permutation τ with k children rooting subtrees of sizes $s_1, \dots, s_k$ from left to right, that has been assigned the set I of consecutive integers and the variable $m = \min(I)$. Then assign sets $I_1, \dots, I_k$ and variables $m_1, \dots, m_k$ to its children so that $m_i = m + \sum_{j:\tau(j)<\tau(i)} s_j$ and $I_i = \{m_i, \dots, m_i + s_i - 1\}$. At the end of this process, every leaf is labeled by an integer m and a set $I = \{m\}$. The signed integer assigned to such a leaf is then either m if the leaf has label $+$ in T or $-m$ if it has label $-$. Notice that the sets I assigned to the nodes of T actually correspond to the strong intervals of σ_T, ensuring that the above mapping is a bijection.

□

Recall that simple permutations are the permutations that have no non-trivial common interval, and are used here as quotient permutations of prime nodes. The enumeration of simple permutations was investigated in [2]. The authors prove that this enumerative sequence is not P-recursive and there is no known closed formula for the number of simple permutations of a given size. Nonetheless they are able to compute a complete asymptotic expression for the number of simple permutations of size n.

Theorem 9 (Albert *et al.* [2])**.** *Let s_n be the number of simple permutations of size n. Then*

$$s_n = \frac{n!}{e^2}\left(1 - \frac{4}{n} + \frac{2}{n(n-1)} + \mathcal{O}\left(\frac{1}{n^3}\right)\right) \text{ when } n \to \infty. \tag{1}$$

3.2 Average shape of strong interval trees

A *twin* in a strong interval tree is a vertex of degree 2 such that each of its two children is a leaf. Thus, a twin is a linear vertex.

Let us notice that all results in this section apply both to signed permutations and unsigned permutations: the two mains reasons for it are that the definition of intervals in a permutation ignores the signs of the elements, and that 2^n signed permutations are associated to any unsigned permutation σ of size n, and this number does not depend on σ.

We first state the main result of this section.

Theorem 10. *Asymptotically, with probability* 1*, a random permutation of size n has a strong interval tree of the form:*

- *the root is a prime vertex;*
- *every child of the root is either a leaf or a twin.*

Moreover, the probability distribution of the number k of twins is given by: $P(k) = \frac{2^k}{e^2 k!}$. Consequently, the expected number of twins is 2*.*

Before proving this result, we can notice that it overlaps with previous results on the expected number of common intervals in permutations. In their paper introducing the problem of computing the common intervals of a permutation [32], Uno and Yagiura showed that the expected number of common intervals of length 2 in a permutation is $2 - 2/n$ and that, for all $\ell > 2$, the expected number of common intervals of size ℓ is 0 for n large enough. This implies immediately our result on the shape of the strong interval tree. Later, Corteel *et al.* showed in [12] that the probability distribution of the number of common intervals of size 2 follows a Poisson law, with mean 2, a result already proved by Kaplanski, in relation with runs in permutations [22]. A similar result was also proved independently in [34]. Theorem 10 gathers all these results together, expressed in terms of the strong interval tree. Moreover, the proof we give here is new and relies on enumerative results on simple permutations.

The proof of Theorem 10 follows from Lemma 11 below and Theorem 9.

Lemma 11. *If $p_{n,k}$ denotes the number of unsigned* [1] *permutations of size n which contain a common interval I of length k then for any fixed positive integer c:*

$$\sum_{k=c+2}^{n-c} \frac{p_{n,k}}{n!} \in \mathcal{O}(n^{-c}).$$

Proof. The proof of Lemma 11 is essentially identical to Lemma 7 of [2]: We have $p_{n,k} \leq (n-k+1)k!(n-k+1)!$. Indeed, the right-hand side counts the number of quotient permutations corresponding to I (which is $k!$), the possible values of the minimal element of I ($n-k+1$) and the structure of the rest of the permutation with one more element for the insertion of I ($(n-k+1)!$). Only the extremal terms of the sum can have magnitude $\mathcal{O}(n^{-c})$ and the remaining terms have magnitude $\mathcal{O}(n^{-c-1})$. Since there are fewer than n terms the result of Lemma 11 follows. □

[1] For signed permutations, the denominator $n!$ should be replaced by $2^n n!$.

Proof of Theorem 10. Lemma 11 with $c = 1$ gives that the proportion of non-simple permutations with at least one common interval of size greater than or equal to 3 is $\mathcal{O}(n^{-1})$. But permutations whose common intervals are only of size $1, 2$ or n are exactly permutations whose strong interval tree has a prime root and every child is either a leaf or a twin.

Similarly, the number of permutations whose strong interval tree has the form of a prime root with k twins is $s_{n-k}\binom{n-k}{k}2^k$. Given the asymptotic estimate of s_n in Equation (1), we compute the asymptotic estimate for the number of such permutations to be $\frac{n!2^k}{e^2 k!}$, proving Theorem 10. □

This result has an immediate corollary in terms of perfect sorting by reversals: the probability that a signed permutation corresponds to an instance that requires an exponential time computation to be solved tends to 0 as n grows.

Corollary 12. *Algorithm BBCP07 runs in $\mathcal{O}(n\sqrt{n \log n})$ time with probability 1 as $n \to \infty$.*

3.3 Average time complexity of perfect sorting by reversals

Further analysis of the tree family $\mathcal{T}_n$ yields a polynomial bound on the average-time complexity of Algorithm BBCP07 (Theorem 15).

Consider the following sum, which is central in the description of the complexity of the algorithm:

$$P_n = \frac{1}{T_n} \sum_p 2^p T_{n,p}.$$

Here T_n is the number of strong interval trees with n leaves ($T_n = |\mathcal{T}_n| = 2^n n!$ from Theorem 8) and $T_{n,p}$ is the number of such trees with p prime vertices. The key step in the algorithm complexity result is essentially reduced to showing $P_n \in \mathcal{O}(1)$.

As an intermediate step, we find a bound on $U_{n,p}$, the number of *unsigned* permutations of size n whose strong interval trees contain p prime vertices, when $p \geq 2$.

Lemma 13. *The number $U_{n,p}$ of unsigned permutations of size n whose strong interval trees contain p prime vertices with $p \geq 2$ is at most $48\frac{(n-1)!}{2^p}$.*

Proof. We proceed by induction on the number p of prime vertices. The hypothesis is the following:

$$(\mathcal{H}_p) : \forall n, U_{n,p} \leq 48\tfrac{(n-1)!}{2^p}.$$

The hypothesis $(\mathcal{H}_p)$ is trivially true for $n < 3p + 1$, since a tree containing p prime vertices has at least $3p + 1$ leaves. We initiate the proof with $p = 2$ assuming $n \geq 7$. A tree of size n with two prime vertices can always be decomposed, although not uniquely, as a tree T_1 that contains one prime vertex, where one leaf is chosen and expanded by a second tree T_2 with one prime vertex. Hence $|T_1| + |T_2| = n + 1$. Without loss of generality, one can assume that the root of T_2 is its only prime vertex. Recall that the number of trees with one prime vertex with k leaves is at most $k!$, as such trees are in bijection with a subset of unsigned permutations of size k. Hence,

$$\begin{aligned}
U_{n,2} &\leq \sum_{k=4}^{n-3} k!k(n+1-k)! \leq (n+1)! \sum_{k=4}^{n-3} \frac{k}{\binom{n+1}{k}} \\
&\leq \frac{(n+1)!}{\binom{n+1}{4}} \sum_{k=4}^{n-3} k \leq \frac{24(n+1)!}{(n+1)n(n-1)(n-2)} \sum_{k=0}^{n-3} k \\
&\leq \frac{24(n-1)!}{(n-1)(n-2)} \frac{(n-3)(n-2)}{2} \leq 48\frac{(n-1)!}{2^2}
\end{aligned}$$

Let us now suppose $(\mathcal{H}_p)$ true and prove $(\mathcal{H}_{p+1})$. We proceed as before. Indeed, a tree with $p + 1$ prime vertices can be decomposed – not necessarily uniquely – as a tree T_1 with p prime vertices, one leaf

of which is expanded by another tree T_2 with one prime vertex. As explained before, we can assume that $n \geq 3(p+1)+1$. Hence:

$$\begin{aligned}
U_{n,p+1} &\leq \sum_{k=3p+1}^{n-3} U_{k,p} k(n+1-k)! \\
&\leq \frac{48}{2^p} \sum_{k=3p+1}^{n-3} (k-1)! k(n+1-k)! \\
&\leq \frac{48(n+1)!}{2^p} \sum_{k=3p+1}^{n-3} \frac{1}{\binom{n+1}{k}} \\
&\leq \frac{48(n+1)!}{2^p} \left[\frac{1}{\binom{n+1}{n-3}} + \sum_{k=3p+1}^{n-4} \frac{1}{\binom{n+1}{k}} \right] \\
&\leq \frac{48(n+1)!}{2^p} \left[\frac{1}{\binom{n+1}{4}} + (n-4-3p) \frac{1}{\binom{n+1}{5}} \right]
\end{aligned}$$

A straightforward analysis by successive derivations on n shows that $\left[\frac{1}{\binom{n+1}{4}} + (n-4-3\cdot 2)\frac{1}{\binom{n+1}{5}}\right] - \frac{1}{2n(n+1)} \leq 0$ for all $n \geq 10$. Hence, since $p \geq 2$, we deduce that $\left[\frac{1}{\binom{n+1}{4}} + (n-4-3p)\frac{1}{\binom{n+1}{5}}\right] \leq \frac{1}{2n(n+1)}$ for all $n \geq 3(p+1)+1$. This ensures that $U_{n,p+1} \leq \frac{48(n-1)!}{2^{p+1}}$ and concludes the proof. □

In the context of signed permutations, Lemma 13 immediately yields the following result:

Lemma 14. *The number $T_{n,p}$ of signed permutations of size n whose strong interval trees contain p prime vertices with $p \geq 2$ is at most $2^n 48 \frac{(n-1)!}{2^p}$.*

Theorem 15. *Computing a shortest perfect scenario for a random signed permutation can be done with average time complexity bounded by $\mathcal{O}(n\sqrt{n \log n})$.*

Proof. First we bound P_n. For all n, by Lemma 14,

$$\begin{aligned}
P_n &= \frac{\sum_p 2^p T_{n,p}}{T_n} \\
&\leq \frac{\left(T_{n,0} + 2T_{n,1} + \sum_{p=2}^{n} 2^n 48(n-1)!\right)}{T_n} \\
&\leq 3 + \sum_{p=2}^{n} \frac{48}{n} \\
&= 3 + 48(1 - \frac{1}{n})
\end{aligned}$$

Thus, $P_n \in \mathcal{O}(1)$. The average time complexity of Algorithm BBCP07 for permutations of size n is given by the following sum, for some constant C:

$$\frac{C}{T_n} \sum_{p=0}^{n} T_{n,p} 2^p n\sqrt{n \log n} = C\, P_n\, n\sqrt{n \log n}.$$

The result follows since $P_n \in \mathcal{O}(1)$.

□

4 Properties of commuting scenarios

We observed in the previous section that the typical shape of the common interval tree associated to a random permutation is very particular, and it is reasonable to ask if a signed permutation selected uniformly at random adequately represents the expected shape of an evolutionary scenario. Experimentally, the strong interval trees that arise when comparing pairs of mammalian genomes contain few prime nodes, labeled with small, simple permutations. Rather, they contain large subtrees with no prime nodes. These subtrees represent commuting scenarios. At present we are unaware of a weighting operator on signed permutations which correlates to the probability that such a permutation could represent an evolutionary scenario on real data. Indeed, such an operator would greatly aid in determining realistic run-times for algorithms on biological data and other properties of evolutionary scenarios. Towards this goal we begin by investigating the class of strong interval trees with no prime nodes. These correspond to commuting scenarios.

The trees that represent commuting scenarios are particularly well-studied. They fall into the category of *simple varieties of trees*, and as such, many formulas exist to compute quantities such as the asymptotic number of trees with n leaves, and also distributions associated to various tree parameters. Some of these parameters have direct relevance to the evolutionary scenario interpretation. Chapter [17, Section VII.3] is a pedagogical reference for simple varieties of trees and we outline how to derive some key values here.

In the remainder of the section, we prove the following results on parsimonious perfect scenarios sorting a commuting signed permutation of size n, via common interval trees:

1. The asymptotic number of commuting permutations is $2^{n+1} \cdot 0.12\,(5.88)^n\, n^{-3/2}$ (a very typical expression for trees) (Equation 6);

2. The average number of reversals in one of these scenarios is $1.2\,n$ (Theorem 16). This is a consequence of the average number of internal vertices in the tree (Equation 8);

3. The average length of a reversal is $1.05\sqrt{n}$. This is related to the average pathlength of a tree. (Theorem 17)

Additionally, in the proof of Theorem 16, we can determine that 37% of the expected reversals have length 1. This agrees with the observation of a large proportion of short reversals in parsimonious scenarios for bacterial genomes [24].

Finally, a note on convergence. The asymptotic estimates we present converge quickly, even for relatively small n. For example, the estimate given for the number of commuting permutations is correct up to order $O(n^{-5/2})$. In real terms, at $n = 100$ it is within 3% of the real value. The parameters have a similar accuracy. The trees that arise from biological data have on the order of 1000 leaves (see [23] for example), and hence these are very strong estimates.

4.1 Modified Schröder Trees

Let σ be a commuting permutation of size n, equivalently, a signed permutation whose strong interval tree $\mathbf{T}(\sigma)$ has no prime node. Thus, $\mathbf{T}(\sigma)$ is a plane tree with the property that the internal vertices have at least two children, each leaf is signed either $+$ or $-$, and the root is also signed $+$ or $-$ (to indicate whether it is an increasing or a decreasing linear vertex). The signs of the other internal vertices follow unambiguously from the sign of the root, alternating between $+$ and $-$ along each branch of the tree.

Disregarding the signs on the leaves and root, this family of trees is known as *Schröder trees* (entry A001003 in the On-Line Encyclopedia of Integer Sequences [28]), and they are straightforward to analyze.

Let $\mathcal{C}$ be the class of all strong interval trees representing commuting permutations, and let $\mathcal{S}$ be the class of Schröder trees. If C_n and S_n respectively denote the number of trees with n leaves in these two classes, then

$$C_n = 2 \cdot 2^n \cdot S_n.$$

Because of this exact $\{1 : 2^{n+1}\}$ correspondence, we generally first consider the class $\mathcal{S}$ to determine structural properties, and then account for the contribution from the leaves. We remark that $\mathcal{S}$ is a subset of the trees $\mathcal{T}$.

4.2 A specification for $\mathcal{S}$

Like many tree classes, there is a simple recursive description for the class $\mathcal{S}$ of Schröder trees: A tree is either a leaf (denoted $\mathcal{L}$), or an internal vertex with at least two subtrees, all of which are elements of $\mathcal{S}$. A visual representation of this statement is given on Figure 2.

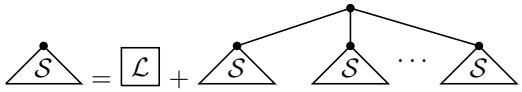

Figure 2: A Schröder tree is decomposed as either a leaf, or an internal vertex with a sequence of subtrees.

We say that the size of a tree $\tau \in \mathcal{S}$ is the number of leaves, and we denote this quantity by $|\tau|$. We shall later consider the number of internal vertices. A leaf is an atomic structure of weight one, and an internal vertex is a neutral structure of weight 0. We translate the above picture description of $\mathcal{S}$ into the following combinatorial equation:

$$\mathcal{S} = \mathcal{L} + \mathrm{Seq}_{\geq 2}(\mathcal{S}), \tag{2}$$

where $\mathrm{Seq}_{\geq 2}(\mathcal{S})$ represents a *sequence* (total order) of at least two trees of $\mathcal{S}$.

4.3 A crash course on decomposable structures

The formalism we are using here is well described in [17]. The main advantage is the direct access to functional equations for the ordinary generating functions. Recall, if S_n is the number trees in $\mathcal{S}$ with n leaves, the ordinary generating function (ogf) is defined as the formal power series $S(z) = \sum_n S_n z^n$. Thus, the series expansion of $S(z)$ begins $S(z) = z + z^2 + 3z^3 + 11z^4 + \cdots$. Many combinatorial actions described on combinatorial classes have companion actions on their generating functions. To summarize, suppose that $\mathcal{A}$ is a combinatorial class with some notion of size, and that A_n is the number of objects of size n. Let $A(z)$ be the series $\sum_n A_n z^n$. If we can express $\mathcal{A}$ in terms of other combinatorial classes, then we can do likewise for its ogf. For example, if $\mathcal{A}$ is the disjoint union of two classes: $\mathcal{A} = \mathcal{B} \uplus \mathcal{C}$, the associated generating functions satisfy the simple relation $A(z) = B(z) + C(z)$. If $\mathcal{A}$ is described using the cartesian product and the size is additive, $\mathcal{A} = \mathcal{B} \times \mathcal{C} = \{(\beta, \gamma) : \beta \in \mathcal{B}, \gamma \in \mathcal{C}\}$, then the ogf satisfy $A(z) = B(z)C(z)$, the usual product for formal power series. Finally, if class $\mathcal{A}$ is a sequence of objects from class $\mathcal{B}$, that is, $\mathcal{A} = \mathrm{Seq}(\mathcal{B}) = \{(\beta_1, \ldots, \beta_k) : 0 \leq k, \beta_i \in \mathcal{B}\}$, then there is the generating function correspondence

$$A(z) = \frac{1}{1 - B(z)}.$$

This is the mere surface of a vast theory rooted in the foundational work of Chomsky and Schutzenberger and their study of algebraic equations related to context free grammars, but significantly advanced and summarized as the theory of decomposable structures in [17].

This is a particularly robust formalism: we can create recursive functional equations, and we can easily pass information about additional parameters. We do both of these here.

4.4 Enumeration formulas

We easily translate the combinatorial description in Eq. (2) into the functional equation[2]

$$S(z) = z + \frac{S(z)^2}{1 - S(z)}. \tag{3}$$

This converts to a simple quadratic equation in $S(z)$. There are two solutions and we choose the one with a Taylor series expansion at 0 with positive integer coefficients, i.e. a generating function solution. This is

$$S(z) = \frac{z + 1 - \sqrt{z^2 - 6z + 1}}{4} = \frac{z+1}{4} - \frac{1}{4}\sqrt{\left(1 - \frac{z}{3 + \sqrt{8}}\right)\left(1 - \frac{z}{3 - \sqrt{8}}\right)}. \tag{4}$$

[2]Here we have used that $\mathrm{Seq}_{\geq 2}(\mathcal{S}) = \mathcal{S} \times \mathcal{S} \times \mathrm{Seq}(\mathcal{S})$.

In order to determine expressions for the asymptotic growth, we follow exactly the procedure outlined in [17, Chapter VI.1], in particular the flow chart of [17, Figure VI.7]. We outline the three main steps of the analysis, but readers interested in further details are referred to this resource.

The first step is to determine the dominant singularity. This is the smallest positive real-valued singularity, which in this case is $3 - \sqrt{8}$.

The second step is to determine the behavior of the function around its dominant singularity, $3 - \sqrt{8}$:

$$S(z) \sim \frac{2-\sqrt{2}}{2} - \frac{1}{2}\sqrt{\sqrt{18}-4}\left(1 - \frac{z}{3-\sqrt{8}}\right)^{\frac{1}{2}} \quad \text{as} \qquad z \sim 3 - \sqrt{8}. \tag{5}$$

We are in a context where asymptotic transfer theorems [17, VI.3] apply, and hence we move to the final step. The approximation of the function near this singularity yields an asymptotic approximation for its coefficients in the Taylor expansion around 0. Roughly, we adapt the following correspondence

$$F(z) \sim \left(1 - \frac{z}{\rho}\right)^{-\alpha} \quad \text{as} \quad z \sim \rho \implies [z^n]F(z) \sim \rho^{-n}\frac{n^{\alpha-1}}{\Gamma(\alpha)}.$$

In this notation, $[z^n]$ extracts the coefficient of z^n in the series expansion of the expression that immediately follows, and Γ is the Gamma function. Using the approximation of $S(z)$ near its dominant singularity from Eq. (5), we deduce

$$S_n = [z^n]S(z) \sim \left(\frac{1}{4}\sqrt{\sqrt{18}-4}\right)(3-\sqrt{8})^{-n}\frac{n^{-3/2}}{\sqrt{\pi}} \sim 0.12\,(5.88)^n\,n^{-\frac{3}{2}}. \tag{6}$$

An asymptotic approximation of the number C_n of signed commuting permutations of size n is obtained by multiplying the above equivalent by 2^{n+1}.

4.5 Tree parameters: A primer

We study the average value of different tree parameters with a common strategy, which we briefly outline here. Let $\chi : \mathcal{S} \to \mathbb{N}$ be an non-negative integer valued function that records some combinatorial property of a Schröder tree, such as the number of internal vertices. The main tool here is the bivariate generating function

$$S(z,u) = \sum_{\tau\in\mathcal{S}} u^{\chi(\tau)} z^{|\tau|} = \sum_{k,n} S_{k,n} u^k z^n,$$

where $S_{k,n}$ is the number of Schröder trees with n leaves, with χ value equal to k. Of course, $S(z) = S(z,1)$ and $S_n = \sum_{k\geq 0} S_{k,n}$. Now, if $\mathbb{E}_n(\chi)$ is the expected value of χ over all objects of size n in $\mathcal{S}$, then by definition

$$\mathbb{E}_n(\chi) = \frac{\sum_{k\geq 0} kS_{k,n}}{\sum_{k\geq 0} S_{k,n}}.$$

We have access to this from the bivariate generating function. Remark,

$$\frac{\partial}{\partial u}S(z,u) = \sum_{k,n} kS_{k,n}u^{k-1}z^n.$$

hence

$$\mathbb{E}_n(\chi) = \frac{[z^n]\frac{\partial}{\partial u}S(z,u)\big|_{u=1}}{[z^n]S(z,1)}.$$

The denominator of this expression is calculated in Equation 6, and in our examples the numerator is a coefficient extraction of an algebraic function of z, hence the three steps described in the previous section apply. Indeed, in our two examples, the dominant singularity is the same as in $S(z)$, $3 - \sqrt{8}$.

This is also a robust approach, and upon considering higher derivatives we can obtain higher moments.

4.6 The average number of internal vertices

We begin by considering the parameter χ equal to the number of internal vertices in a Schröder tree. We can augment the specification in Eq. (2) with a neutral marker μ of weight 0 to tag the internal vertices:

$$\mathcal{S} = \mathcal{L} + \mu \times \mathrm{Seq}_{\geq 2}(\mathcal{S}). \tag{7}$$

We now consider the bivariate generating function $S(z,u)$ where u marks χ, which counts the total number of markers. Eq. (7) translates to a functional equation for $S(z,u)$:

$$S(z,u) = z + u\frac{S(z,u)^2}{1-S(z,u)}.$$

This is solved in a similar manner to $S(z)$:

$$S(z,u) = \frac{z+1-\sqrt{(z+1)^2-4z(u+1)}}{2(u+1)} = z + uz^2 + (u+2u^2)z^3 + \ldots.$$

This expression can be differentiated to determine

$$\frac{\partial}{\partial u}S(z,u)|_{u=1} = \frac{(z-1)^2-(z+1)\sqrt{(z+1)^2-8z}}{8\sqrt{(1-\frac{z}{3+\sqrt{8}})(1-\frac{z}{3-\sqrt{8}})}}.$$

As we remarked above, the singularity analysis follows as before using the singularity $3-\sqrt{8}$. Thus,

$$\frac{\partial}{\partial u}S(z,u)|_{u=1} \sim \alpha \cdot \left(1-\frac{z}{3-\sqrt{8}}\right)^{-1/2} \quad \text{as} \quad z \sim 3-\sqrt{8},$$

where the constant α is determinnd by evaluating the rest of the expression at $z = 3-\sqrt{8}$. The third step, applying the transfer theorem, is the performed. To determine the average, we divide this expression by the asymptotic number of trees, as determined in Equation (6). We simplify the radicals, and obtain

$$\mathbb{E}_n(\chi) = \frac{[z^n]\frac{\partial}{\partial u}S(z,u)\big|_{u=1}}{[z^n]S(z,1)} \sim \frac{3-\sqrt{8}}{3\sqrt{2}-4}\,n \sim \frac{n}{\sqrt{2}}. \tag{8}$$

4.7 The average number of reversals.

Next we use the average number of internal vertices to count the average number of reversals in a scenario. An evolutionary scenario is obtained from tree in $\mathcal{S}$ by signing the root, and the leaves. Each internal vertex, except the root, represents a reversal. A leaf represents a reversal if and only if it has a sign different from the sign of its parent.

The number of internal vertices is a good first approximation for the number of reversals. Asymptotically, since the average number of vertices is a linear function of n, subtracting by one to account for the root has little or no effect.

In order to account for the reversals at the leaves, we remark that for any tree in $\mathcal{S}$ of size n, we consider all 2^n possible ways of assigning signs to the leaves, and from this symmetry we deduce that on average this adds $n/2$ reversals, all of length 1.

We put all of these pieces together in the following theorem.

Theorem 16. *The asymptotic average number of reversals in a parsimonious perfect scenario of a random signed commuting permutation of size n is $n/\sqrt{2} + n/2 = \frac{1+\sqrt{2}}{2}\,n$. On average there are $n/2$ reversals of length 1.*

4.8 The average length of a reversal

The length of a reversal in a scenario is equal to the size (number of leaves) in the corresponding subtree of the common interval tree. Again, our analysis first estimates by studying a parameter on unsigned trees, $\mathcal{S}$, and then tunes by considering the reversals of size one represented by signs on the leaves.

Our analysis is guided by the study of a related parameter called pathlength that frequently makes a cameo appearance when trees are used to study sorting algorithms.

Let $\Psi(\tau)$ be the sum of the subtree sizes for all subtrees in $\tau \in \mathcal{S}$. Examining Figure 2, we can formulate a recursive description of $\Psi(\tau)$. Consider $\tau \in \mathcal{S}$. Either τ is a single leaf, in which case $\Psi(\tau) = 0$, or the root has m children, labeled left to right by $\tau_1, \ldots, \tau_m$. To compute $\Psi(\tau)$, we sum the sizes of the subtrees of each child of the root, and then add the size of the entire tree, which of course is the sum of the sizes of the children. This is written

$$\Psi(\tau) = \sum_{j=1}^{m} \big(\Psi(\tau_j) + |\tau_j|\big).$$

A tree parameter that satisfies such a relationship is an *additive parameter* and writing the corresponding functional equation is straightforward. We mark the parameter Ψ by the variable v in the bivariate generating function $S(z, v)$:

$$S(z, v) = \sum_{\tau \in \mathcal{S}} v^{\Psi(\tau)} z^{|\tau|} = z + v^2 z^2 + (v^3 + 2v^5)z^3 + \ldots \tag{9}$$

This parameter is identical to the pathlength parameter, and the steps from the generating function to the equation are all well-explained in [17, Section III.5]. We derive the functional equation

$$S(z, v) = z + \frac{S(vz, v)^2}{1 - S(vz, v)}. \tag{10}$$

Rather than solve for $S(z, v)$, it is easier to solve for $\frac{\partial}{\partial v} S(z, v)|_{v=1}$ directly by differentiating Eq. (10) with respect to z and v, and setting $v = 1$ in the resulting equations. This leads to two equations in two unknowns. Using the notation $S_v(z) = \frac{\partial}{\partial v} S(z, v)|_{v=1}$, $S_z(z) = \frac{\partial}{\partial z} S(z, v)|_{v=1}$, and recalling $S(z, 1) = S(z)$, the counting ordinary generating function for $\mathcal{S}$, this leads to the system

$$S_v(z) = \frac{S(z)(2 - S(z))(S_z(z)z + S_v(z))}{(1 - S(z))^2}$$
$$S_z(z) = 1 + \frac{S(z)S_z(z)(2 - S(z))}{(1 - S(z))^2}.$$

We solve $S_v(z)$ in terms of $S(z)$:

$$S_v(z) = \frac{zS(z)(2 - S(z))(1 - S(z))^2}{(1 - 4S(z) + 2S(z)^2)^2} = 2z^2 + 13z^3 + 80z^4 + \ldots$$

This is an explicit expression to which we apply singularity analysis to determine an expression for the coefficient of z^n:

$$[z^n] \frac{\partial}{\partial v} S(z, v)|_{v=1} = [z^n] S_v(z) \sim \frac{\sqrt{2}}{16}(3 - \sqrt{8})^{-n}.$$

This value approximates the sum of the sizes of all subtrees of all trees in $\mathcal{S}$. The average value of Ψ is the quotient

$$\mathbb{E}_n(\Psi) = \frac{[z^n] S_v(z)}{[z^n] S(z)} \sim \left(\frac{\sqrt{2}}{16}(3 - \sqrt{8})^{-n}\right) \left(\frac{1}{4}\sqrt{\sqrt{18} - 4}(3 - \sqrt{8})^{-n} \frac{1}{\sqrt{\pi n^3}}\right)^{-1}$$
$$= \frac{\sqrt{\pi}}{4\sqrt{3 - \sqrt{2}}}\, n^{\frac{3}{2}} \sim 1.27\, n^{\frac{3}{2}}$$

To get the expected sum of the lengths of the reversals of a parsimonious perfect scenario for a random commuting permutation, we consider adjustments that occur for each tree in $\mathcal{S}$, so we can add them directly

to this value. For each tree we remove the size of the whole tree (n) since we do not count this as a reversal, and we also add the average contribution of the reversals of size 1 ($n/2$). These two adjustments do not affect the asymptotic growth since $n^{\frac{3}{2}}$ dominates n for large n.

To determine the the average length, we now divide by the average number of reversals, which we determined to be $\frac{1+\sqrt{2}}{2}n$ in Theorem 16 We summarize these results in the following theorem.

Theorem 17. *The average length of a reversal in a parsimonious perfect scenario for a random signed commuting permutation of size n is asymptotically*

$$\frac{\sqrt{\pi}}{2(1+\sqrt{2})\sqrt{3-\sqrt{2}}}\sqrt{n} \sim 1.054\sqrt{n}.$$

5 Conclusion

Summary Perfect sorting by reversals, although an intractable problem, is very likely to be solved in polynomial time for random signed permutations, under the uniform distribution. This result relies on a study of the shape of a random strong interval tree that shows that asymptotically such trees are mostly composed of a large prime vertex at the root and small subtrees. We were also able to give precise asymptotic results for the expected lengths of a parsimonious perfect scenario and of a reversal of such a scenario for random commuting permutations. Our results were obtained using techniques of enumerative and analytic combinatorics.

Discussion on our results. Recently, several works have investigated average properties of combinatorial objects related to genomic distance computation, such as the breakpoint graph [29, 33, 35], conserved segments [31] or adjacencies and common intervals [12, 34]. The motivation for such works can be twofold. One can be interested in the expected behavior of some algorithms, such as in [29], that shows that the most intricate part of the theory of sorting by reversals (clearing hurdles and fortress) is not required on uniform random permutations. Our results on the average complexity of computing a parsimonious perfect scenario belong to this family of results. In other cases, one can be interested in the expected properties of an evolutionary scenario for random genomes [33, 35]. This allows, given real data, to assess the significance of the comparison of a pair of genomes and to compute statistical tests measuring the evolutionary signal left: intuitively, if a scenario between two real genomes looks like a scenario between random genomes, one can make the hypothesis that there is little to no evolutionary signal left in the considered pair of real genomes. Our results on commuting permutations are of such nature.

The fact that computing a parsimonious perfect scenario requires polynomial time on the average is mostly a theoretical result, that completes the complexity analysis of the problem. Indeed, real data sets (pairs of genomes or genome segments) are in general not expected to define strong interval trees with a large number of prime nodes (see [23] for example). So the algorithms described in [4, 6] were already known to be efficient on real data sets.

It should however be noted that our results on the expected shape of a strong interval tree, and in particular on the number of prime vertices, generalizes previous results on conserved adjacencies and common intervals in permutations [32, 12, 34]. They could form the basis for a deeper study of the expected shape of the strong interval tree, parametrized by the number of common adjacencies or of prime nodes. Also, the combinatorial specification of the class of strong interval trees opens the way to random generation algorithms [18] of trees with some prescribed structure (such as the number or maximum degree of prime nodes), that we outline in the paragraph below on future research. This might allow to study by simulations the expected properties of a perfect scenario between pairs of genomes defining strong interval trees with a prescribed structure. Such way to assess the significance of features of a hypothetical scenario between real genomes is clearly of practical interest.

In the same vein, the strong interval trees obtained in comparing pairs of mammalian genomes for example [23] contain very few prime nodes, and then contain large subtrees that represent commuting permutations; these subtrees can then be compared to the expected properties of random commuting permutations to point at genome segments whose evolution is significantly non-random.

Future research. There exists several more general models of genome rearrangements [15]. Among them, the more general is based on an operation called *Double-Cut-and-Join* (DCJ for short) that models reversals and several other types of rearrangements. The notion of perfect DCJ scenarios has been studied in [5] and has the intriguing property that instances that were hard to solve for reversals can be solved in polynomial time in the DCJ model and conversely. It would then be interesting to compare the average time complexity of perfect sorting by DCJ to the results we describe in the present work.

We could, modulo the labeling of the prime nodes by simple permutations, easily describe $\mathcal{T}_n$ using a grammar in the combinatorial calculus described in [17]. This would give access to enumerative and structural information, when paired with the generating function for simple permutations. Generating these trees directly, i.e. without first generating the corresponding permutation, remains an interesting open problem, that seems to be well suited to Boltzmann random generation techniques [14].

Also, PQ-trees are a natural family of trees that are both related to common intervals of permutations [7] and used in comparative genomics [23]. Investigating average properties of PQ-trees is a natural extension of the work presented here.

More generally, average properties of the many families of combinatorial objects that appear in comparative genomics models and algorithms is an almost completely open field, that contains many challenging problems and deserve being investigated.

6 Acknowledgments

Authors MM and CC acknowledge funding support from the Discovery Grant program of the Natural Science and Engineering Research Council (Canada). MB and DR are supported by the ANR projects GAMMA (BLAN07-2_195422) and MAGNUM (2010_BLAN_0204).

References

[1] M. H. Albert and M. D. Atkinson. Simple permutations and pattern restricted permutations. *Discrete Math.*, 300:1–15, 2005.

[2] M. H. Albert, M. D. Atkinson, and M. Klazar. The enumeration of simple permutations. *J. Integer Seq.*, 6:03.4.4, 2003.

[3] S. Bérard, A. Bergeron, and C. Chauve. Conservation of combinatorial structures in evolution scenarios. In *Comparative Genomics 2004*, volume 3388 of *LNCS/LNBI*, pages 1–14, 2004.

[4] S. Bérard, A. Bergeron, C. Chauve, and C. Paul. Perfect sorting by reversals is not always difficult. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 4:4–16, 2007.

[5] S. Bérard, A. Chateau, C. Chauve, C. Paul, and E. Tannier. Computation of perfect DCJ rearrangement scenarios with linear and circular chromosomes. *J. Comput. Biol.*, 16:1287–1309, 2009.

[6] S. Bérard, C. Chauve, and C. Paul. A more efficient algorithm for perfect sorting by reversals. *Inform. Proc. Letters*, 106:90–95, 2008.

[7] A. Bergeron, C. Chauve, F. de Montgolfier, and M. Raffinot. Computing common intervals of k permutations, with applications to modular decomposition of graphs. *SIAM J. Discrete Math.*, 22:1022–1039, 2008.

[8] A. Bergeron, J. Mixtacki, and J. Stoye. *Mathematics of Evolution and Phylogeny*, chapter The inversion distance problem. Oxford University Press, 2005.

[9] K.S. Booth and G.S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity. *J. Comput. Syst. Sci.*, 13:335–379, 1976.

[10] M.D. Braga, M.F. Sagot, C. Scornavacca, E. Tannier. Exploring the solution space of sorting by reversals, with experiments and an application to evolution. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 5:348–356, 2008.

[11] B.-M. Bui-Xuan, M. Habib, and C. Paul. Revisiting T. Uno and M. Yagiura's Algorithm. In *ISAAC 2005*, volume 3827 of *LNCS*, pages 146–155, 2005.

[12] S. Corteel, G. Louchard and R. Pemantle. Common Intervals in Permutations. *Discrete Math. Theor. Comput. Sci.*, 8:189–214, 2006.

[13] Y. Diekmann, M.-F. Sagot, and E. Tannier. Evolution under reversals: Parsimony and conservation of common intervals. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 4:301–109, 2007.

[14] P. Duchon, P. Flajolet, G. Louchard, and G. Schaeffer. Boltzmann Samplers for the Random Generation of Combinatorial Structures. *Combin. Probab. Comput.*, 13:577-625, 2004.

[15] G. Fertin, A. Labarre, I. Rusu, E. Tannier, and S. Vialette. *Combinatorics of Genome Rearrangements*. MIT Press, 2009.

[16] M. Figeac and J.-S. Varré. Sorting by reversals with common intervals. In *WABI 2004*, volume 3240 of *LNCS/LNBI*, pages 26–37, 2004.

[17] P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

[18] P. Flajolet, P. Zimmerman and B. Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoretical Computer Science*, 132:1-2, pages 1–35, 1994.

[19] S. Hannenhalli and P. A. Pevzner. Transforming cabbage into turnip: Polynomial algorithm for sorting signed permutations by reversals. *J. ACM*, 46:1–27, 1999.

[20] S. Heber and J. Stoye. Finding all common intervals of k permutations. In *CPM 2001*, volume 2089 of *LNCS*, pages 207–218, 2001.

[21] L. Ibarra. Finding pattern matchings for permutations. *Inform. Proc. Letters*, 61:293–295, 1997.

[22] I. Kaplansky. The asymptotic distributions of runs of consecutive elements. *Annals of Mathematical Statistics*, 16:200-203, 1945.

[23] G. M. Landau, L. Parida, and O. Weimann. Gene proximity analysis across whole genomes via PQ trees. *J. Comput. Biol.*, 12:1289–1306, 2005.

[24] J.-F. Lefebvre, N. El-Mabrouk, E. R. M. Tillier, and D. Sankoff. Detection and validation of single gene inversions. In *Bioinformatics*, pages i190–i196, 2003.

[25] B.M.E. Moret, J. Tang, and T. Warnow. *Mathematics of Evolution and Phylogeny*, chapter Reconstructing phylogenies from gene-content and gene-order data. Oxford University Press, 2005.

[26] Q. Peng, P.A. Pevzner, and G. Tesler. The fragile breakage versus random breakage models of chromosome evolution. *PLoS Comput. Biol.*, 2:e14, 2007.

[27] D. Sankoff. Edit distances for genome comparisons based on non-local operations. In *CPM 1992*, volume 644 of *LNCS*, pages 121–135, 1992.

[28] N. J. A. Sloane. The on-line encyclopedia of integer sequences, 2007. Published electronically at `www.research.att.com/~njas/sequences/`.

[29] K. M. Swenson, Y. Lin, V. Rajan, and B. M. E. Moret. Hurdles hardly have to be heeded. In *RECOMB-CG 2008*, volume 5267 of *LNCS/LNBI*, pages 241–251, 2008.

[30] E. Tannier, A. Bergeron, and M.-F. Sagot. Advances on sorting by reversals. *Discrete Appl. Math.*, 155:881–888, 2007.

[31] G. Tesler. Distribution of Segment Lengths in genome Rearrangements. *Elec. J. Combinat.* 15:R105, 2008.

[32] T. Uno and M. Yagiura. Fast Algorithms to Enumerate All Common Intervals of Two Permutations. *Algorithmica*, 26:290–309, 2000.

[33] W. Xu. The distribution of distances between randomly constructed genomes: Generating function, expectation, variance and limits. *J. Bioinform. Comput. Biol.*, 6:23–36, 2008.

[34] W. Xu, B. Alain, and D. Sankoff. Poisson adjacency distributions in genome comparison: multichromosomal, circular, signed and unsigned cases. *Bioinformatics*, 24:i146–i152, 2008.

[35] W. Xu, C. Zheng, and D. Sankoff. Paths and cycles in breakpoint graph of random multichromosomal genomes. *J. Comput. Biol.*, 14:423–435, 2007.